7. Scoreboard

Program Name: Scoreboard.java Input File: scoreboard.dat

Since you are currently competing in UIL CS, you are already aware of how the scoring procedure works. The programming portion consists of 12 programming questions, each worth 60 points. A problem is considered solved by a team if any of that team's submission for the problem is accepted. A penalty score of 5 points is deducted for each incorrect submission before an accepted submission. This means that if a team submits multiple incorrect solutions for a problem but never correctly solves it, they are not penalized. Whereas a team that submits a problem correctly on the second attempt only receives 55 points. The number of points that a team receives for correctly solving a problem should never drop below 0.

Input

The first line of input contains T, the number of test cases that follow. This is followed by a blank line. There is also a blank line between consecutive test case inputs. Each test case represents a snapshot of the judging queue, containing entries for some or all of the teams 1 through 25 solving problems 1 through 12.
The first line of each test case contains N, the number of entries in the judging queue. The following N lines of input each represent a single entry. Each entry is in the format: team number, problem number, and the *team problem status*, where the *status* can be `A`, `I`, `C`, `U`, `E`. These symbols stand for Accepted submission, Incorrect submission, Clarification request, Unjudged submission, and Erroneous submission. Clarification requests, unjudged submissions, and erroneous submissions do not affect the scoring.

Output

The output for each test case will be the scoreboard, sorted by the total points earned using the criteria described above. Each line of output will contain the team number, the number of problems solved by that team, and the total score accumulated by the team. Since not all teams may be actively participating, only display the teams that have made submissions (accepted, incorrect, unjudged, and erroneous). If two or more teams are tied, they are displayed in order of increasing team numbers.

The output of two consecutive test cases will be separated by a blank line.

Constraints

```
1 <= T <= 10
1 <= N <= 25
```

Sample Input File

```
3

5
1 2 I
3 1 A
1 2 C
1 2 A
1 1 A

6
5 2 U
5 1 C
14 0 C
14 4 E
5 0 C
2 1 C

9
21 3 I
15 2 A
21 3 C
2 1 A
21 12 A
21 7 U
21 3 A
15 5 E
6 1 E
```

Example Output to Screen

```
1 2 115
3 1 60

5 2 115
14 1 60

21 2 115
2 1 60
15 1 60
6 1 0
```